

Reduce false positives!
False Positive Detection using CodeDx AI/ML function

誤検出を削減！
CodeDx AI/ML機能を用いた、False Positive検出

Dec 3 2019

チャンネル・マネージメント・ソリューションズ株式会社
代表取締役社長 マーク・グライス

AI/ML Assisted False Positive Identification

(AI/MLがアシストする誤検知識別)

**New functionality
currently in development**
(現在開発中の機能)

DAST \Leftrightarrow SAST Hybrid Correlation

(DAST \Leftrightarrow SASTハイブリッド相関)

**Recently introduced
new functionality**
(最近導入された新しい機能)

AI/ML Assisted False Positive Identification… (AI / MLアシストによる誤検知の識別…)



AI: Artificial Intelligence

ML: Machine Learning

Machine learning is an application of **artificial intelligence** that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

機械学習は人工知能のアプリケーションであり、明示的にプログラムしなくても、システムが経験から自動的に学習し改善する機能を提供します。

- Learn from the data being processed and improve over time as more data is processed.
処理中のデータから学び、より多くのデータが処理されるにつれて時間とともに改善されます。
- Will “evolve” differently depending on the data that is learned from to become more efficient/accurate in the specific environment it is learning from.
学習元のデータに応じて異なる「進化」を行い、学習元の特定の環境でより効率的/正確になります。

AI/ML Assisted False Positive Identification...

(AI / MLアシストによる誤検知の識別...)



R&D

Release
Q1 2020

※2020年春
リリース予定

Code Dx AI/ML assisting False Positive (FP) identification for classifying findings of users.

(ユーザの調査結果を分類する為のFP識別を支援するCode DxのAI/ML)

4Eval.

Real
Projects

The ML engine (early β level) has been applied in some customer environments for real world validation and evaluation.

(MLエンジンは、実際の検証と評価のためにいくつかの顧客の環境で適用中)

96%

Average
Accuracy

Accuracy of FP prediction for newly reported findings has been above 90% from 4 large production environment deployments.

(新たに報告された調査結果のFP予測の精度は90%を超えています)



AI/ML Assisted False Positive Identification… (AI / MLアシストによる誤検知の識別…)



Code Dx can further reduce the time and effort to go through large amounts of findings by providing assistance to the user in automatically identifying findings which are most likely FPs.

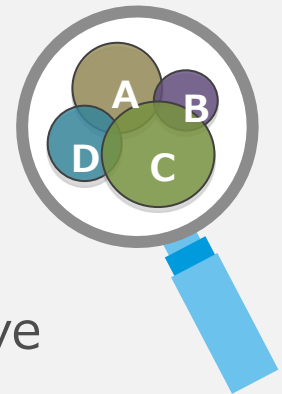
Code Dxは、FPである可能性が高い検出結果を自動的に識別する際に、ユーザーを支援することで、大量の検出結果を調べる時間と労力をさらに削減できます。

- With hundreds or thousands of findings coming from the suite of testing tools being used drilling down to understanding which of those findings are important is a **difficult and time consuming task**

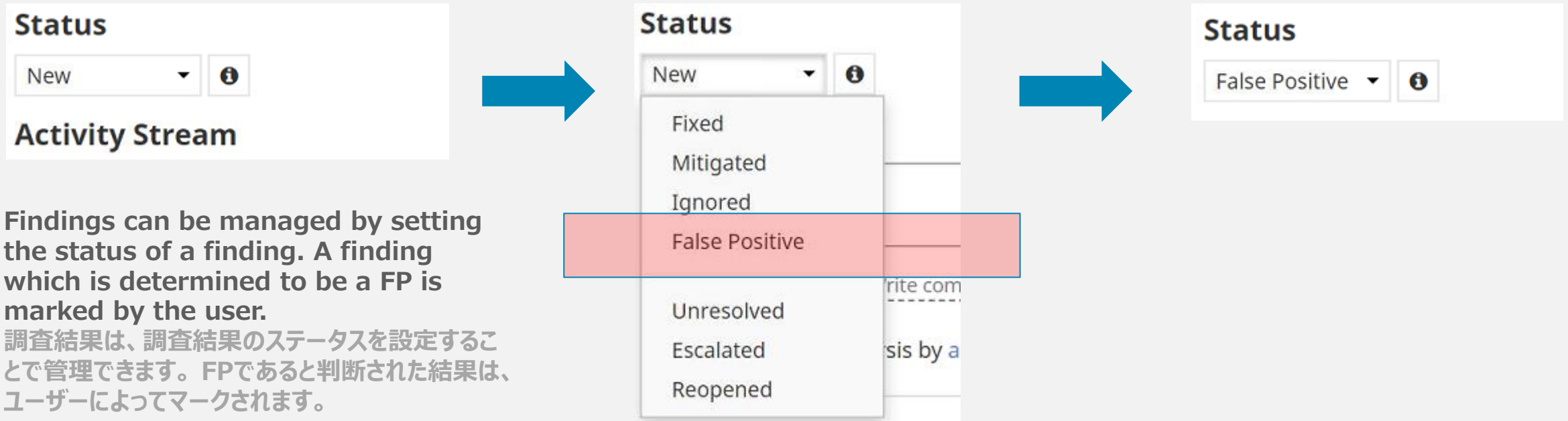
使用されている一連のテストツールから得られる数百または数千の調査結果をドリルダウンして、それらの調査結果のどれが重要であるかを理解することは困難で時間のかかる作業です。

- Being able to quickly surface the findings which are critical/important and separate these from the vast amounts of less important findings allows teams to **focus remediation resources where they are most needed** and will be the most effective

使用されている一連のテストツールから得られる数百または数千の調査結果をドリルダウンして、それらの調査結果のどれが重要であるかを理解することは困難で時間のかかる作業です。



AI/ML Assisted False Positive Identification… (AI / MLアシストによる誤検知の識別…)



Findings can be managed by setting the status of a finding. A finding which is determined to be a FP is marked by the user.

調査結果は、調査結果のステータスを設定することで管理できます。FPであると判断された結果は、ユーザーによってマークされます。

By default all FP findings are “removed” (i.e. hidden) from the working set of Findings.

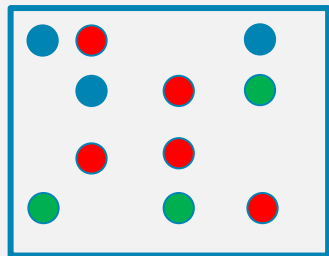
デフォルトでは、FPの検出結果はすべて、検出結果のワーキングセットから「削除」（つまり非表示）されます。

AI/ML Assisted False Positive Identification...

(AI / MLアシストによる誤検知の識別...)

Each finding is comprised of a set of data points which form a unique signature (or fingerprint) for that specific finding.

各調査結果は、その特定の調査結果の一意的署名（または指紋）を形成するデータポイントのセットで構成されます。

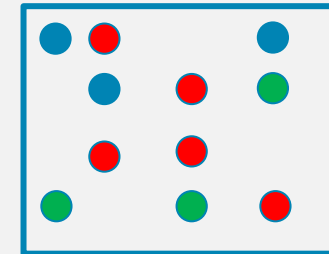
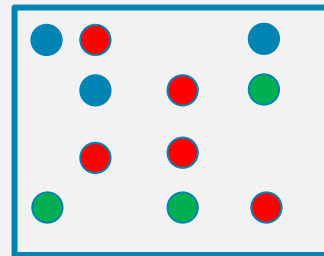


The data points of a finding can be represented by a "matrix" which encapsulates the specific aspects of that finding producing a unique signature.

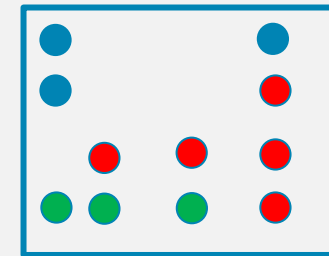
調査結果のデータポイントは、その調査結果の特定の側面をカプセル化して「一意的署名」を生成する「マトリックス」で表すことができます。

Findings can be compared
調査結果を比較できます

Reference

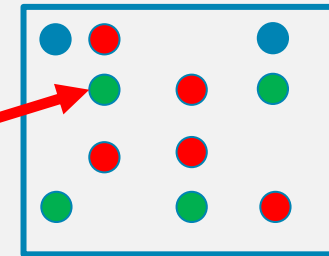


Match
(Exact)



Not a
Match

Small difference only
わずかな違いのみ



Match
(within
scope)

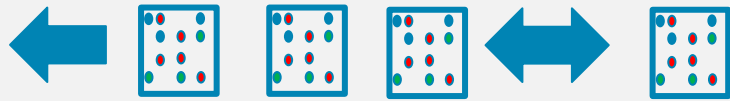
AI/ML Assisted False Positive Identification...

(AI / MLアシストによる誤検知の識別...)

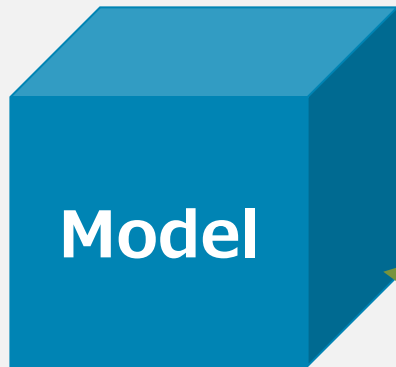


History: Management of findings in the past provides a history to study and learn from

履歴：過去の調査結果の管理は、調査および学習するための履歴を提供します



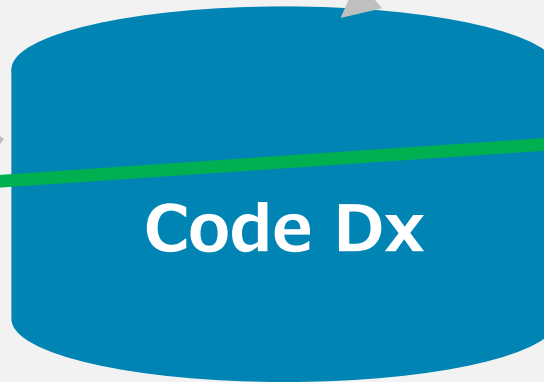
1



3
Generate signature



4
Compare



New Finding

2

Status
False Positive ▾ ⓘ

As findings are classified as FP the model is enhanced and refined
調査結果がFPとして分類されると、モデルは強化され洗練されます。

7

5

Predict if likely a FP & mark the new finding accordingly
FPである可能性が高いかどうかを予測し、それに応じて新しい結果をマークする

6

Overriding predicted FP can provide feedback loop to ML engine to refine the model
予測されたFPをオーバーライドすると、MLエンジンにフィードバックを提供してモデルを改良できます

DAST \Leftrightarrow SAST Hybrid Correlation

Code Dx Hybrid Correlation (HC) enables the results from DAST tools and processes to be correlated with the results from SAST tools and processes.

Code Dx Hybrid Correlation (HC) により、DASTツールおよびプロセスの結果をSASTツールおよびプロセスの結果と相関させることができます。

- Greater visibility into how findings (vulnerabilities) may occur or be exploited in production product
製品で検出結果（脆弱性）がどのように発生するか、または悪用されるかについての可視性が向上
- Assist in identifying test cases for those findings
それらの調査結果のテストケースの特定を支援する
- Show location at source code level where actual run-time error condition occurred
実際の実行時エラー状態が発生した場所をソースコードレベルで表示する

アプリケーション開発時のテスト方法	テスト範囲	主なテストツール名
DAST (Dynamic Application Security Testing)	Runtime testing/verification	E.g. Vex Vulnerability Explorer, AppScan, etc...
SAST (Static Application Security Testing)	Source code analysis/verification	E.g. CheckMarx CxSAST, Klocwork, QAC, etc...

Maximizing testing coverage requires the utilization of available testing/verification technologies. Dynamic analysis and static analysis are two well recognized methods.

テスト範囲を最大化するには、利用可能なテスト/検証技術を利用する必要があります。
動的分析と静的分析は、よく知られている2つの方法です。

- Dynamic analysis (runtime testing) uncovers actual (real) vulnerabilities and error conditions and can find issues not detectable by static analysis
動的分析（実行時テスト）は、実際の（実際の）脆弱性とエラー状態を明らかにし、静的分析では検出できない問題を見つけることができます
- Static analysis (source code verification) provides exhaustive and deep analysis uncovering many potential vulnerabilities and error conditions and highlighting the location at the code level
静的分析（ソースコード検証）は、多くの潜在的な脆弱性とエラー状態を明らかにし、コードレベルで場所を強調する徹底的かつ詳細な分析を提供します。
- Code Dx HC can show which possible SA findings were actually reached during runtime testing, allowing developers to prioritize findings to fix based on actual business criticality.
Code Dx HCは、ランタイムテスト中に実際に到達した可能性のあるSAの結果を表示できるため、開発者は、実際のビジネス上の重要性に基づいて修正する検出結果に優先順位を付けることができます。

Problem: It is often very difficult and time consuming to track down where in the source code to go to fix a finding found during runtime testing

問題：ランタイムテスト中に見つかった結果を修正するためにソースコード内のどこに行くかを追跡することは、多くの場合非常に困難で時間がかかります。

Code Dx provides a tracing agent which can be used to track actual code execution during runtime testing and use this information to match error conditions encountered with corresponding static analysis findings.

Code Dxは、実行時テスト中に実際のコード実行を追跡し、この情報を使用して、対応する静的分析結果で発生したエラー条件と一致させるために使用できるトレースエージェントを提供します。

- Supported for Java and JSP applications

JavaおよびJSPアプリケーションでサポート

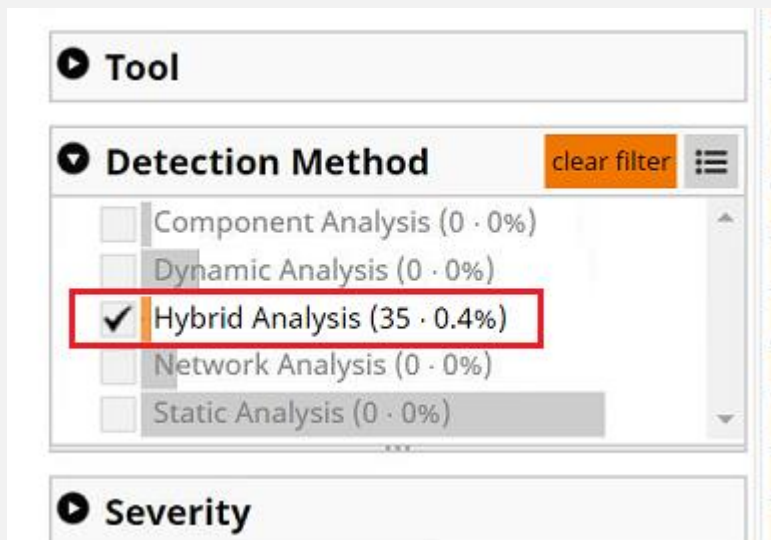
In addition to the correlation of DAST and SAST findings, the instrumentation based hybrid analysis also allows developers to see how much of the application was reached during run time testing (DAST coverage).

DASTとSASTの結果の相関関係に加えて、インストルメンテーションベースのハイブリッド分析により、開発者はランタイムテスト中にどれだけのアプリケーションに到達したかを確認できます（DASTカバレッジ）

- Understand test plan/test case coverage and weaknesses

テスト計画/テストケースの範囲と弱点を理解する

Hybrid Correlation ...



New filter to focus on findings which have DAST<-> SAST correlation

DAST <-> SAST相関を持つ検出結果に焦点を当てる新しいフィルター

ID	Issue Type	Active Results	Source Code Location
22930	Information Exposure	6 active results from Nessus, ZAP	/examp...
22911	Information Exposure	4 active results	Checkmarx / JavaMediumThreat / XSRF s/j...
22872	Information Exposure	5 active results	Checkmarx / JavaMediumThreat / XSRF s/c...
22775	Information Exposure	4 active results	Checkmarx / JavaMediumThreat / XSRF s/j...
22297	Information Exposure	4 active results	Checkmarx / JavaMediumThreat / XSRF s/c...
22271	Information Exposure	4 active results	ZAP / Absence of Anti-CSRF Tokens s/c...
22183	Information Exposure	4 active results	ZAP / Absence of Anti-CSRF Tokens s/...
22022	Information Exposure	5 active results	ZAP / Absence of Anti-CSRF Tokens pgo...
21909	Cryptographic Issue	10 active results	ZAP / Absence of Anti-CSRF Tokens ack
5276	Cross-Site Request Forgery (CSRF)	12 active results	ZAP / Absence of Anti-CSRF Tokens CSRF.ja...

- 12 findings, from both SAST (CheckMarx) and DAST (ZAP) related to a single root cause in the source code
ソースコードの単一の根本原因に関連するSAST (CheckMarx) とDAST (ZAP) の両方からの12の調査結果。
- Drill down to actual location in source code
ソースコードの実際の場所にドリルダウンします。

```
99
100 statement.setInt(1, count++);
101 statement.setString(2, title);
102 statement.setString(3, message);
103 statement.setString(4, s.getUserName());
104 statement.setString(5, this.getClass().getName());
105 statement.execute();
106
107 } catch (Exception e)
108 {
109     s.setMessage("Could not add message to database");
```

Hybrid Correlation ...

WebGoat-HybridAgent Instrumentation Dashboard + New Ana

9,966 Findings Last analyzed 8/29/2019 at 8:38:40 PM Analysis took 2 min, 35 sec

Covered vs non-covered classes/methods

Coverage: Total vs. custom (non third-party/lib) code

Display coverage at class or method level
クラスまたはメソッドレベルでカバレッジを表示する

Select which part(s) of the application to view
表示するアプリケーションの部分を選択します

WebGoat-HybridAgent » Instrumentation Total code coverage: 6% Custom code coverage: 43%

Application Inventory	method count	% Coverage
- Classes	1824	0%
- org.owasp.webgoat	1824	0%
-<self>	20	0%
+ .application	14	0%
+ .controller	12	0%
+ .lessons	1314	0%
+ .service	41	0%
+ .servlets	3	0%
+ .session	371	0%
+ .util	49	0%
+ JARs	71K	0%
+ JSPs	64	0%

Code Treemap

Treemap item size

- based on number of bytecode instructions
- based on number of lines of code

Treemap level of detail

- show methods
- show classes

Unnamed Trace Session

from: 3 months ago
duration: 22 min
coverage: 3%

No active sessions
[Start one](#)

New Tracer Agent

Select single or multiple traces (test sessions)
単一または複数のトレースを選択します (テストセッション)

